L Number	Hits	Search Text	DB	Time stamp
3	535	(700/17).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:13
4	266	(700/18).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:14
5	450	(700/2).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:14
6	29036	mask$2 near5 (data or value)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:17
9	16	((700/2).CCLS.) and (mask$2 near5 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:15
7	16	((700/17).CCLS.) and (mask$2 near5 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:15
8	13	((700/18).CCLS.) and (mask$2 near5 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:16
10	2995	mask$2 near5 (data or value) and cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:17
11	4	((700/17).CCLS.) and (mask$2 near5 (data or value) and cach$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:17
12	1	((700/18).CCLS.) and (mask$2 near5 (data or value) and cach$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:18
13	2	((700/2).CCLS.) and (mask$2 near5 (data or value) and cach$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:18
-	2269	I/O adj processor	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 13:13
-	71	(I/O adj processor) with cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 13:13
-	52	((I/O adj processor) with cach$3) and @ad<=20010725	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:08
-	6	(I/O adj processor) and (force near5 mask)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:05

-	29	(I/O adj processor) and (forced near5 data)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:07
-	108	(I/O adj processor) and (mask$2 near5 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:29
-	86	(I/O adj processor) and (mask$2 near2 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:08
-	86	((I/O adj processor) and (mask$2 near2 (data or value))) and @ad<=20010725	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:08
-	0	force adj on adj mask	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:09
-	0	force adj on adj mask$1	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:09
-	2387	force near5 mask$1	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:10
-	6	(I/O adj processor) and (force near5 mask$1)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:20
-	136	I/O adj value	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:26
-	1	(I/O adj value) near5 forced	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:21
-	1	(I/O adj value) near5 (forced or mask$2)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:22
-	6	(I/O adj value) and cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:27
-	3	(I/O adj forc$3e) and cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:28
-	8	(I/O adj forc$3) and cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:28
-	29036	(mask$2 near5 (data or value))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:30

-	238	((mask$2 near5 (data or value))) with cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:30
-	10	((mask$2 near5 (data or value))) with cach$3 with I/O	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:42
-	67	infrequently near2 (changed or altered) near3 data	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:43
-	5	(infrequently near2 (changed or altered) near3 data) with cach$3	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:44
-	837	(711/113).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:44
-	1443	(711/118).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:55
-	870	(711/147).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:55
-	904	(710/1).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	861	(710/5).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	180	(710/23).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	326	(710/48).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	85	(710/49).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	225	(710/262).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:56
-	136	(712/224).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	0	("19and24").PN.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57

-	57	((mask$2 near5 (data or value))) and ((711/118).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	40	((mask$2 near5 (data or value))) and ((711/147).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	42	((mask$2 near5 (data or value))) and ((710/1).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	34	((mask$2 near5 (data or value))) and ((710/5).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	9	((mask$2 near5 (data or value))) and ((710/23).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	10	((mask$2 near5 (data or value))) and ((710/28).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	19	((mask$2 near5 (data or value))) and ((710/48).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	33	((mask$2 near5 (data or value))) and ((710/49).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	49	((mask$2 near5 (data or value))) and ((710/262).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	80	((mask$2 near5 (data or value))) and ((712/224).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:57
-	24	((mask$2 near5 (data or value))) and ((711/113).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:59
-	0	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((711/118).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:59
-	0	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((711/147).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 14:59
-	0	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/28).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:00
-	0	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/49).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:00

-	0	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((712/224).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:00
-	1	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((711/113).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:00
-	4	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/1).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:00
-	3	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/5).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:01
-	1	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/23).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:01
-	1	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/48).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:02
-	1	(I/O adj processor) and (((mask$2 near5 (data or value))) and ((710/262).CCLS.))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/11/09 15:02

Your search matched **4** of **1091947** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

forced <near> (data <or> value) <sentence> cache [Search]

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

1 **Fault-containment in cache memories for TMR redundant processor systems**
Chung-Ho Chen; Somani, A.K.;
Computers, IEEE Transactions on , Volume: 48 , Issue: 4 , April 1999
Pages:386 - 397

[Abstract] [PDF Full-Text (412 KB)] **IEEE JNL**

---

2 **An efficient wavelet-based compression method for volume renderii**
Kim, T.; Shin, Y.;
Computer Graphics and Applications, 1999. Proceedings. Seventh Pacific Conference on , 5-7 Oct. 1999
Pages:147 - 156

[Abstract] [PDF Full-Text (924 KB)] **IEEE CNF**

---

3 **Improved adaptive replacement algorithm for disk caches in HSM systems**
Hahn, U.; Dilling, W.; Kaletta, D.;
Mass Storage Systems, 1999. 16th IEEE Symposium on , 15-18 March 1999
Pages:128 - 140

[Abstract] [PDF Full-Text (824 KB)] **IEEE CNF**

---

4 **Memory hierarchy management for iterative graph structures**
Al-Furaih, I.; Ranka, S.;
Parallel Processing Symposium, 1998. 1998 IPPS/SPDP. Proceedings of the Fi Merged International...and Symposium on Parallel and Distributed Processing 1998 , 30 March-3 April 1998
Pages:298 - 302

PORTAL

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ◉ The ACM Digital Library ○ The Guide

forced <near> I/O <near> (data <or> value) <sentence> cac

SEARCH

THE ACM DIGITAL LIBRARY

Advanced Search

Search Tips

Enter words, phrases or names below. Surround phrases or full names with double quotation marks.

Search within Results: **82,715** found

forced <near> I/O <near> (data <or> value) <sentence> cache

Clear result set

SEARCH

Desired Results:

must have **all** of the words or phrases

"forced value", I/O processor

must have **any** of the words or phrases

must have **none** of the words or phrases

Name or Affiliation:

Authored by: ◉ all ○ any ○ none

Edited by: ◉ all ○ any ○ none

Reviewed by: ◉ all ○ any ○ none

Only search in:*

○ Title ○ Abstract ○ Review ◉ All Information

SEARCH

*Searches will be performed on all available information, including full text where available, unless specified above.

ISBN / ISSN: ◉ Exact ○ Expand

DOI: ◉ Exact ○ Expand

SEARCH

Published:

By: ◉ all ○ any ○ none

In: ◉ all ○ any ○ none

Since:

Month Year

Before:

Month Year

As: Any type of publication

Conference Proceeding:

Sponsored By:

Conference Location:

Conference Year:

yyyy

SEARCH

Classification: (CCS) ☐ **Primary Only**

Classified as: ◉ all ○ any ○ none

Subject Descriptor: ◉ all ○ any ○ none

Keyword Assigned: ◉ all ○ any ○ none

Results must have accessible:

☐ Full Text ☐ Abstract ☐ Review

SEARCH

PORTAL
US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

+"forced value", I/O processor

SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **forced value I/O processor**

Found **25** of **82,715 searched out of 123,523.**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 25

Result page: **1** 2 next

Relevance scale

1 Monte Carlo techniques in code optimization
Dean Jacobs, Jan Prins, Peter Siegel, Kenneth Wilson
October 1982 **Proceedings of the 15th annual workshop on Microprogramming**

Full text available: pdf(545.73 KB)

Additional Information: full citation, abstract, references, citings, index terms

Effective optimization of FPS Array Processor assembly language (APAL) is difficult. Instructions must be rearranged and consolidated to minimize periods during which the functional units remain idle or perform unnecessary tasks. Register conflicts and branches cause complications. Deterministic algorithms to arrange instructions traditionally use complex heuristics which are tailored to specific inputs. A non-deterministic approach can be simpler and effective on a large class of inputs. T ...

2 Performance evaluation of the Orca shared-object system
Henri E. Bal, Raoul Bhoedjang, Rutger Hofman, Ceriel Jacobs, Koen Langendoen, Tim Rühl, M. Frans Kaashoek
February 1998 **ACM Transactions on Computer Systems (TOCS)**, Volume 16 Issue 1

Full text available: pdf(179.39 KB)

Additional Information: full citation, abstract, references, citings, index terms, review

Orca is a portable, object-based distributed shared memory (DSM) system. This article studies and evaluates the design choices made in the Orca system and compares Orca with other DSMs. The article gives a quantitative analysis of Orca's coherence protocol (based on write-updates with function shipping), the totally ordered group communication protocol, the strategy for object placement, and the all-software, user-space architecture. Performance measurements for 10 parallel applications ill ...

Keywords: distributed shared memory, parallel processing, portability

3 WaveScalar
Steven Swanson, Ken Michelson, Andrew Schwerin, Mark Oskin
December 2003 **Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture**

Full text available: pdf(228.98 KB)
Publisher Site

Additional Information: full citation, abstract, citings, index terms

Silicon technology will continue to provide an exponential increasein the availability of raw transistors. Effectively translatingthis resource into application performance, however,is an open challenge. Ever increasing wire-delay relativeto switching speed and the exponential cost of circuit complexitymake simply scaling up existing processor designs futile.In this paper, we present an alternative to superscalardesign, WaveScalar. WaveScalar is a dataflow instructionset architecture and executi ...

4 Revised report on the algorithmic language scheme
J Rees, W Clinger
December 1986 **ACM SIGPLAN Notices**, Volume 21 Issue 12
Full text available: pdf(4.06 MB) Additional Information: full citation, citings, index terms

5 ProtoMol, an object-oriented framework for prototyping novel algorithms for molecular dynamics
Thierry Matthey, Trevor Cickovski, Scott Hampton, Alice Ko, Qun Ma, Matthew Nyerges, Troy Raeder, Thomas Slabach, Jesús A. Izaguirre
September 2004 **ACM Transactions on Mathematical Software (TOMS)**, Volume 30 Issue 3
Full text available: pdf(907.26 KB) Additional Information: full citation, abstract, references, index terms

ProtoMol is a high-performance framework in C++ for rapid prototyping of novel algorithms for molecular dynamics and related applications. Its flexibility is achieved primarily through the use of inheritance and design patterns (object-oriented programming). Performance is obtained by using templates that enable generation of efficient code for sections critical to performance (generic programming). The framework encapsulates important optimizations that can be used by developers, such as parall ...

Keywords: Fast electrostatic methods, incremental parallelism, molecular dynamics, multigrid, multiple time-stepping integration, object-oriented framework.

6 Kernel Korner: Dynamic Kernels - Modularized Device Drivers
March 1996 **Linux Journal**
Full text available: html(29.61 KB) Additional Information: full citation, index terms

7 Haptic sculpting of dynamic surfaces
Frank Dachille, Hong Qin, Arie Kaufman, Jihad El-Sana
April 1999 **Proceedings of the 1999 symposium on Interactive 3D graphics**
Full text available: pdf(1.15 MB) Additional Information: full citation, references, citings, index terms

8 A placement algorithm for array processors
Dah-Juh Chyan, Melvin A. Breuer
June 1983 **Proceedings of the 20th conference on Design automation**
Full text available: pdf(648.80 KB) Additional Information: full citation, abstract, references, citings, index terms

In this report a concurrent pairwise exchange placement algorithm executing on an array processor is presented. Two force functions and their effects are discussed. The oscillation phenomenon caused by the concurrent computation is investigated and some solutions are suggested. A design for the array processor is presented along with a complexity analysis which indicates that this algorithm is O(N2) faster than a conventional sequential placement

algorithm if N&t ...

Keywords: Array-processor, Oscillation, Pairwise exchange, Placement, VLSI

9 Abstract routing of logic networks for custom module generation
S. T. Healey, W. J. Kubitz
October 1987 **Proceedings of the 24th ACM/IEEE conference on Design automation**

Full text available: pdf(845.98 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes a switchbox-type router for custom VLSI module generation as performed by a module planner. A module is decomposed into abstract cells consisting of global routes and Boolean functional specifications. Each abstract cell is given to a cell synthesizer which generates the circuit layout and through-the-cell routing. Abstract routing for a module planner is in some sense similar to switchbox routing to the degree that all of the routes are generated inter ...

10 System-Level Synthesis of Application Specific Systems using A* Search and Generalized Force-Directed Heuristics
Chunho Lee, Miodrag Potkonjak, Wayne Wolf
November 1996 **Proceedings of the 9th International Symposium on System Synthesis**

Full text available: pdf(822.59 KB) Publisher Site Additional Information: full citation, abstract

This paper presents a system-level approach to the synthesis of multi-task, hard real-time applications. The goal is to select a set of off-the-shelf processors with minimal cost while satisfying timing constraints. Our approach has three design phases: resource allocation, assignment, and scheduling. With the observation that the resource allocation is a search for a set of processors which requires the minimum cost, we adopted A* search based technique. For assignment we use a variation of the ...

Keywords: Hard Real-Time, System-Level Synthesis, Search Techniques

11 A technique for QoS-based system partitioning
Johnson S. Kin, Chunho Lee, William H. Mangione-Smith, Miodrag Potkonjak
January 2000 **Proceedings of the 2000 conference on Asia South Pacific design automation**
Full text available: pdf(92.35 KB) Additional Information: full citation, references

12 Revised5 report on the algorithmic language scheme
N. I. Adams, D. H. Bartley, G. Brooks, R. K. Dybvig, D. P. Friedman, R. Halstead, C. Hanson, C. T. Haynes, E. Kohlbecker, D. Oxley, K. M. Pitman, G. J. Rozas, G. L. Steele, G. J. Sussman, M. Wand, H. Abelson
September 1998 **ACM SIGPLAN Notices**, Volume 33 Issue 9
Full text available: pdf(4.44 MB) Additional Information: full citation, citings, index terms

13 Compiler technology for parallel machines: Language support for data parallelism in pointer based dynamic data structures
Pankaj Kumar

October 1993 **Proceedings of the 1993 conference of the Centre for Advanced Studies on Collaborative research: distributed computing - Volume 2**
Full text available: pdf(634.64 KB) Additional Information: full citation, abstract, references

To date, most of the work on language support for data parallelism has been largely limited to static data structures such as arrays. In 'C' or C++, the use of pointers and dynamic memory allocators provide the ability to handle more complex data structures whose size and shape may vary over the course of the computation. There are two main issues with regards to supporting data parallelism on pointerbased dynamic data structures without having the programmer worry about synchronization, ...

14 Fault diagnosis based on effect-cause analysis: An introduction
Miron Abramovici, Melvin A. Breuer
June 1980 **Proceedings of the 17th conference on Design automation**
Full text available: pdf(691.37 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents the basic concepts of a new fault diagnosis technique which has the following features: 1) is applicable to both single and multiple faults, 2) does not require fault enumeration, 3) can identify faults which prevent initialization, 4) can indicate the presence of nonstuck faults in the D.U.T., 5) can identify fault-free lines in the D.U.T. Our technique, referred to as effect-cause analysis, does not require a fault dictionary and it is not based on com ...

15 Borel sets and circuit complexity
Michael Sipser
December 1983 **Proceedings of the fifteenth annual ACM symposium on Theory of computing**
Full text available: pdf(534.71 KB) Additional Information: full citation, abstract, references, citings, index terms

It is shown that for every k, polynomial-size, depth-k Boolean circuits are more powerful than polynomial-size, depth-(k−1) Boolean circuits. Connections with a problem about Borel sets and other questions are discussed.

16 Efficient visualization of physical and structural properties in crash-worthiness simulations (case study)
Sven Kuschfeldt, Thomas Ertl, Michael Holzner
October 1997 **Proceedings of the 8th conference on Visualization '97**
Full text available: pdf(592.85 KB) Publisher Site Additional Information: full citation, references, index terms

17 DAB: interactive haptic painting with 3D virtual brushes
Bill Baxter, Vincent Scheib, Ming C. Lin, Dinesh Manocha
August 2001 **Proceedings of the 28th annual conference on Computer graphics and interactive techniques**
Full text available: pdf(10.82 MB) Additional Information: full citation, abstract, references, citings, index terms

We present a novel painting system with an intuitive haptic interface, which serves as an expressive vehicle for interactively creating painterly works. We introduce a deformable, 3D brush model, which gives the user natural control of complex brush strokes. The force feedback enhances the sense of realism and provides tactile cues that enable the user to better manipulate the paint brush. We have also developed a bidirectional, two-layer paint model that, combined with a palette interface ...

Keywords: *Human Computer Interaction, deformable brush model, haptics, painting systems*

18 A test synthesis approach to reducing BALLAST DFT overhead
Douglas Chang, Mike Tien-Chien Lee, Malgorzata Marek-Sadowska, Takashi Aikyo, Kwang-Ting Cheng
June 1997 **Proceedings of the 34th annual conference on Design automation - Volume 00**
Full text available: pdf(168.52 KB) Publisher Site
Additional Information: full citation, abstract, references, index terms

In this paper, we present a test synthesis approach which integratesBALLAST (BALAnced structure Scan Test) withan enhanced test point insertion (TPI) algorithm to functionallyscan the flip-flops chosen by BALLAST.BALLASTis an attractive partial scan technique in that it offers combinationalATPG efficiency while promising to reduce full scanoverhead.However, the practical problem with BALLASTis it typically requires more scan flip-flops than other partialscan techniques.The TPI enhancements enabl ...

19 The combination of scheduling, allocation, and mapping in a single algorithm
Richard J. Cloutier, Donald E. Thomas
January 1991 **Proceedings of the 27th ACM/IEEE conference on Design automation**
Full text available: pdf(792.19 KB)
Additional Information: full citation, abstract, references, citings, index terms

We present a single high level synthesis algorithm that schedules the operations of a data dependence graph, allocates the necessary hardware, and maps the operations to specific functional units. This is achieved by extending the global analysis approach developed for force-directed scheduling to include individual module instances. This new algorithm should be applicable to any behavioral synthesis system that schedules operations from a data dependence graph.

20 Interaction and VR: A model for efficient and accurate interaction with elastic objects in haptic virtual environments
Dan C. Popescu, Michael Compton
February 2003 **Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia**
Full text available: pdf(260.21 KB) Additional Information: full citation, abstract, references

This paper describes a method of modelling real-time interactions with elastic 3D objects represented by finite element models, which is particularly suitable for haptic virtual environments. The assumption we make is that the area of interaction of the external forces on the object is small. Our method provides a physically based solution and only requires the precomputation of the inverse of the stiffness matrix. It can be naturally coupled with a technique of local multiresolution collision d ...

Keywords: deformable objects, finite element method, haptics, linear elasticity, virtual reality